FACULTY WORKING PAPER 93-0150

An Integrated Approach to Product Design and Process Selection - Revised

Dilip Chhajed
Department of Business Administration
University of Illinois

Narayan Raman
Department of Business Administration
University of Illinois

Bureau of Economic and Business Research
College of Commerce and Business Administration
University of Illinois at Urbana-Champaign

BEBR

FACULTY WORKING PAPER NO. 93-0150

College of Commerce and Business Administration

University of Illinois at Urbana-Champaign

August 1993

An Integrated Approach to Product Design and Process Selection - Revised

Dilip Chhajed
Narayan Raman

An Integrated Approach to Product Design and Process Selection

Dilip Chhajed

Narayan Raman

Department of Business Administration
University of Illinois at Urbana-Champaign
Champaign, Illinois 61820

November 1992
Revised July 1993

ABSTRACT

The conventional approach to new product design involves a sequential flow of decisions. At the first step, the marketing function determines the product profile by specifying the required product attributes based upon consumer research. Manufacturing is then expected to determine the appropriate processes capable of delivering these attributes efficiently. In practice, such a sequential approach leads to substantial delays in product introduction and frequent product redesigns long after it has been introduced. Increasing competitive pressure necessitates an alternative, integrated approach in which the product design and the process selection decisions are made simultaneously.

This study models such an integrated approach for the objective of maximizing the producer's incremental profit as a nonlinear integer program. Process selection is merged with product design by explicitly considering the alternative processes available, and the associated fixed and variable costs, for providing an attribute at a given level. The overall decision involves simultaneously determining the levels at which each attribute will be present in the new product as well as the processes used for providing these attribute levels. In addition, we treat product price as an extrinsic decision variable. In view of the problem complexity, we propose a heuristic solution method that decomposes the overall problem into the product design and the process selection subproblems; the solution algorithm essentially iterates between these two subproblems.

Computational studies indicate that the suggested algorithm performs effectively with respect to the optimal solution. Furthermore, the integrated approach to product and process design provides substantial improvement over the sequential approach.

1 INTRODUCTION

Manufacturing and service firms need to introduce new products frequently in order to remain competitive. Product design involves the consideration of the needs and preferences of individual market segments, products offered by the competition, as well as the company's own existing products in view of the possible cannibalization. The marginal profitability of the new product clearly depends upon the fixed and variable production costs in addition to the incremental net revenue generated. Thus, the process selection decision plays an important role in determining the overall success of the new product. The traditional approach to product design and process selection, also known as "over the wall approach" and "relay race", involves a sequential flow of decisions. Marketing function determines the "product profile" by specifying the required product attributes. Subsequently, the production function determines the processes capable of delivering these attributes. While such a delineation of decisions appears to be logical conceptually, in practice it leads to substantial delays in product introduction, bureaucracy, high cost, and frequent redesigns long after it has been introduced.

Increasing competitive pressure has necessitated a significant reduction in the lead time required for introducing new products. This trend has generated growing interest in an integrated approach in which both product and process are designed concurrently. Such an integrated approach has been operationalized in a variety of ways such as simultaneous engineering (Evans 1988), concurrent engineering (Brazier and Leonard 1990), quality function deployment, and the house of quality (Hauser and Clausing 1988). In particular, the house of quality provides a good framework for an objective analysis as well as a tool for integrating the product design process. While there is substantive empirical evidence suggesting the potential of the integrated approach, the literature on modeling this problem is relatively sparse.

The bulk of the existing literature on product design focuses exclusively on consumer preferences. Two approaches to this problem have been proposed – Multidimensional Scaling (MDS) and Conjoint Analysis. In the MDS approach, each customer's ideal product is represented as a point in a continuous multiattribute space whose dimensions are differentially weighted across customers. The relative preference of any customer for a given product is inversely related to the "distance" of the product from the customer's ideal point in this space. Product selection is either deterministic, first-choice or probabilistic in which the probability of a particular product being chosen is related to its distance from the customer's ideal point. Product design optimization based on the MDS approach was introduced by Shocker and Srinivasan (1974), has since been pursued by several other researchers (see, for example, Zufreyden 1979, Hauser and Simmie 1981, Gavish, Horsky and Srikanth 1983, Sudharshan, May and Shocker 1987, and Eliashberg and Manrai 1989). The objectives commonly considered include maximizing incremental profit and maximizing market share. Albers (1976) and Sudharshan, May and Gruca (1988) extend this research to a product line.

In the typical Conjoint Analysis approach, which is the methodology adopted in this study, each attribute

is specified at one of several potential discrete levels. In most formulations, the overall utility of the product for a given customer is given by the sum of the idiosyncratic *part worth* of the level at which each attribute is active. The bulk of this research employs a deterministic first-choice selection criterion. A customer chooses the product with highest utility; alternatively, if price is considered as an extrinsic attribute, the product which provides the maximum surplus - utility net of price, is selected. Optimal product design based on conjoint analysis was first developed by Zufreyden (1977) for the objective of maximizing market share which has since been extended by Green, Carroll and Goldberg (1981). Other objectives that have been studied in the context of conjoint analysis include maximizing seller's return (Green and Kreiger 1985, Kohli and Krishnamurti 1987, Dobson and Kalish 1988, McBride and Zufreyden 1988) and maximizing buyer's welfare (Green and Kreiger 1985). In their recent work, Kohli and Sukumar (1990) extend the approach proposed by Kohli and Krishnamurti (1987) to include all three objectives for introducing a product line.

Much of the existing research on product design does not explicitly consider the cost of providing the product. As Johnson (1974) notes, these studies "..assume that these versions (of product) are all feasible from a manufacturing and pricing standpoint, that we could produce any one of them, and we wish to choose the "best" version." Significant exceptions include Dobson and Kalish (1988, 1993) who consider both fixed and variable manufacturing costs of each product, and also consider price to be an extrinsic variable. They construct a general model for selecting a line of products from a prespecified set of candidate products, and construct a two stage heuristic to obtain the approximate solution. Eriksen and Berger (1987) develop a quadratic programming model that considers fixed and attribute level-specific variable costs for designing fitness centers to be located at European airports. Green and Kreiger (1992) describe the use of the SIMOPT conjoint analysis based model for optimally designing a liquid dietary supplement. Their study incorporates direct manufacturing and distribution cost specific to each attribute level. Chakravarty and Baum (1992) formulate a product line design model that includes process selection considerations. For given product prices, this model posits the decision as a nonlinear programming problem in the general case. Chakravarty and Baum construct problem instances to highlight the interaction between certain marketing and manufacturing variables, such as the impact of process flexibility on the optimal product mix. Cook and Gill (1993) formulate a product design problem that involves cost (comprising fixed investment cost and variable manufacturing costs) tradeoffs among various design alternatives.

A parallel body of literature considers product development purely from the manufacturing perspective. The emphasis of the bulk of work done in this line of research addresses the importance of incorporating production costs and other manufacturing considerations while developing engineering design specifications; in so doing, they do not directly consider customer requirements. Various approaches proposed to facilitate this process are Design for Manufacture (DFM, Boothroyd and Dewhurst 1988, Stoll 1988), Design for Assembly (Boothroyd 1988), and Design for Analysis (Suri and Shimizu 1989). In presenting a successful application of DFM to the manufacture of Polaroid cameras, Ulrich et al. (1993) note that an important

next step is "an integrated view of product development that includes the strategic and market implications of the design as well as production costs. Conventional DFM methodologies may inhibit this integration because of their focus on production costs."

The model developed in this study addresses the introduction of a *single* new product for the objective of maximizing the producer's incremental profit. The significant point of departure of this work from previous research on product design lies in providing a methodology for integrating the product design and process selection decisions. Unlike the Dobson-Kalish model, we consider the process selection decision explicitly. Furthermore, we address each attribute level individually, and build the optimal (or near-optimal) product profile directly from these levels. Process selection is integrated into the product design problem by explicitly considering the alternative processes available, and the associated fixed and variable costs, for providing an attribute at a given level. The overall decision involves simultaneously determining the levels at which each attribute will be present in the new product(s) as well as the processes used for providing these attribute levels.

This work differs from Chakravarty and Baum (1992) in three important aspects. First, we model the product selection criterion *individually* for each customer based upon his or her surplus maximization. Second, akin to the Dobson-Kalish model, we treat product price as an extrinsic decision variable. Third, we propose a solution approach for linking the product design and process selection decisions. While the cost structures used in Eriksen and Berger (1987), Green and Kreiger (1992) and Cook and Gill (1993) are similar to ours in that they relate to each attribute level, theses studies do not consider the process dependence of these costs, and hence, do not explicitly address the process selection decision.

This model, together with the proposed solution approach, is a first step towards building a decision support tool to aid the coordination of different functional areas within an organization through better communication. The incorporation of manufacturing costs into the model enables us to consider the tradeoff between using the processes currently available within the company and buying new equipment. Because the new product is likely to share a number of common features with existing products (Srinivasan and Shocker 1979), the current manufacturing facility may be capable of providing certain attribute levels with negligible increase in fixed costs, while on the other hand, new equipment has the potential of reducing variable costs. Furthermore, the model provides a basis for evaluating the economic worth of process flexibility. For example, consider the decision of selecting between a (relatively inflexible) process that can provide only a limited number of attributes, and an alternative process capable of providing a wider range of attributes, albeit less efficiently. The outcome of this decision clearly depends upon the mix of attributes under consideration, as well as the demand volume generated by the customers who switch to the new product. Similarly, whether or not an attribute will be present at a given level depends upon the associated manufacturing cost, in addition to the additional revenue that it will generate.

The paper is organized as follows. §2 discusses the major characteristics and assumptions of the model. We formulate the problem as a nonlinear mixed integer program, and show that it is NP-hard. In §3, we present an iterative heuristic solution procedure that decomposes the problem into product design and process selection subproblems. We describe an example problem in §4, and evaluate the performance of the suggested heuristic in §5. In §6, we discuss how the proposed approach can serve as a decision support tool. §7 summarizes the main results of this study. Proofs of mathematical results stated in the paper are given in Appendix 1.

2 The Model

The model is based on the conjoint analysis approach to estimating consumer preferences. The product to be designed is comprised of K attributes. The attributes considered in this study are the *physical*, as opposed to *perceptual*, characteristics of a product. A given attribute $k \in \mathcal{K}$ is realized in the product at exactly one of J_k discrete levels. Some of these attributes can be options which occur at two levels – 'present' and 'absent'. The set of levels for attribute k is denoted by $\mathcal{J}_k$. A product profile is denoted by $\mathbf{X} = (X_1, \ldots, X_K)$, where $X_k = (x_{1k}, \ldots, x_{J_k k})^T$, and $x_{jk} = 0$ or 1.

$\mathcal{I} = (1, 2, \ldots, I)$ denotes the set of customers. A given customer $i \in \mathcal{I}$ represents either an individual or a segment. The weight p_i is a measure of segment population and (annualized) product purchase frequency for customer i. Consistent with conjoint analysis models of product design (see, for example, Green and Kreiger 1985, 1992, Kohli and Krishnamurti 1987, and Kohli and Sukumar 1990), we assume that part worths w_{ijk} corresponding to each level j of an attribute k can be determined for each for customer i. In keeping with these models, we also assume that the utility U_i of any product $\mathbf{X}$ for customer i, $i = 1, \ldots, I$ is the sum of the part worths of individual attributes; i. e., $U_i = \sum_k \sum_j w_{ijk} x_{jk}$. We also assume that the customer employs a deterministic first-choice selection criterion based on maximizing his/her surplus. Therefore, given the utility U_i, price π and surplus $u_i (= U_i - \pi)$ of the new product, customer i will switch if $u_i \geq u_i^0$ where u_i^0 refers to the surplus of the product currently purchased by customer i. [The utility U_i^0 and the price π_i^0 for customer i's existing product are similarly defined.] l_i denotes the marginal contribution made by customer i to the firm; it is nonzero only if i is a customer of one of the existing products made by the firm. We do not model the competitive reaction to product design decision made by the producer. Thus, the situations considered here are that of no competition or passive competition in the sense of Dobson and Kalish (1988).

The set of candidate processes is denoted by $\mathcal{M}$, and $|\mathcal{M}| = M$. Each process $m \in \mathcal{M}$ has an annualized fixed cost of f_m, and a variable cost of v_{jkm} for producing attribute k at level j. In this paper, a process denotes an entity that can deliver an attribute at a given level. Thus, a process can be a single workcenter, a manufacturing cell or even an entire plant; it could also denote a subcontractor. We assume that each process has unlimited capacity. While, as we discuss later in §3.2, it is possible to include these constraints

in the overall solution approach, doing so introduces discontinuities that obscure useful insights that are otherwise available from a basic model.

Before presenting the model, we preview some of its important characteristics in order to better define its scope. Product design usually involves several iterations through multiple stages (Urban, Hauser and Dholakia 1987). A typical modeling representation of product design includes, in particular, the stages of i) product functionality specification based on customer requirements, ii) determination of engineering characteristics, and iii) selection of manufacturing processes. In this model, we do not consider the first two stages separately; instead, we assume that product functionality and engineering design considerations have been taken into account while developing the various attribute levels and their associated costs. This is relatively straightforward for products in which there is a one-to-one correspondence between product attributes and engineering characteristics. For example, in a hair brush, the handle can be treated as an attribute, with the various designs in which it can be provided as the individual levels.

In other, more complex products, specifying attributes and attribute levels, will likely require a number of iterations and tradeoffs involving both customer requirements and engineering design considerations. This can be done through consultation and interviews with marketing and design personnel (Page and Rosenbaum 1987, Green and Kreiger 1992). In this model, an attribute is general enough to also represent a design subsystem within the product; the individual attribute levels then correspond to the different design alternatives available for each subsystem. Physical prototypes of these design alternatives can then be used to determine the attribute-level part worths (Winter 1993). In such cases, the attribute level variable cost corresponds to the *optimal* configuration of the design alternative in terms of the manufacturing costs. In permitting the variable costs to be process-specific, the model additionally allows these configurations to be process-dependent[1]. A case in point is Page and Rosenbaum's (1987) application of conjoint analysis for redesigning food processors at Sunbeam, Inc. Several of the attributes considered – motor power, number of processing blades, bowl type, bowl size, bowl shape, type of feed tube pusher, size of feed tube, etc. are essentially design subsystems. Individual attribute levels, such as the three levels of regular, heavy duty and professional, correspond to the design alternatives available for providing motor power. Part worths for each alternative was determined by providing their sketches to the various respondents.

The model is quite general in that no structure is imposed on the part worths which can be defined arbitrarily in the multiattribute space for the individual customers. The model allows for the cannibalization of existing customers by the new product. Similarly, in regard to the process, no restriction is placed on the values that

[1] This paradigm of treating a product as a collection of its subsystems, and evaluating each design alternative of these subsystems in terms of their fixed and variable costs is consistent with the practice followed in many companies (Cook 1993). We note that approaches such as House of Quality (Hauser and Clausing 1988) can be used to determine the various design alternatives; while DFM can help in selecting the optimal configuration for each alternative (Ulrich et al. 1993). Also see Eriksen and Berger 1987, and Cook and Gill 1993 for alternative approaches to attribute level cost determination.

the fixed and variable costs can take. The formulation also permits process flexibility, i. e., a given process can be used for producing more than one attribute. Finally, note that although the model considers the introduction of a single product rather than a product line, its usefulness may not be limited in many practical applications. As Dobson and Kalish (1993) observe, most firms involved in new product introduction face market-related uncertainties and limited availability of R&D and marketing resources. It is desirable for these firms to consider sequential introduction of products focusing on one product at a time. Because the model permits the presence of other, competing products in the market made by the same manufacturer, it can be gainfully used in such situations.

The integrated product design and process selection problem (**PDPS**) is formulated as the following mathematical program:

PDPS

$$Z = \max \sum_{i \in \mathcal{I}} (\pi p_i - l_i) s_i - \sum_{m \in \mathcal{M}} f_m q_m - \sum_{i \in \mathcal{I}} p_i s_i \left(\sum_{m \in \mathcal{M}} \sum_{k \in \mathcal{K}} \sum_{j \in \mathcal{J}_k} v_{jkm} z_{jkm} \right) \tag{1}$$

subject to

$$\sum_{k \in \mathcal{K}} \sum_{j \in \mathcal{J}_k} w_{ijk} x_{jk} - \pi - u_i^0 \leq B s_i \quad \forall i \tag{2}$$

$$\pi - \sum_{k \in \mathcal{K}} \sum_{j \in \mathcal{J}_k} w_{ijk} x_{jk} + u_i^0 \leq B(1 - s_i) \quad \forall i \tag{3}$$

$$\sum_{j \in \mathcal{J}_k} x_{jk} = 1 \quad \forall k \tag{4}$$

$$\sum_{m \in \mathcal{M}} z_{jkm} \geq x_{jk} \quad \forall j, k \tag{5}$$

$$z_{jkm} \leq q_m \quad \forall j, k, m \tag{6}$$

$$\pi \geq 0; s_i, z_{jkm}, x_{jk}, q_m \in (0, 1) \quad \forall i, j, k, m \tag{7}$$

where B is a large number; $s_i = 1$ if customer i switches to the new product, 0 otherwise; $x_{jk} = 1$ if attribute k is provided at level j in the new product, 0 otherwise; $q_m = 1$ if process m is selected, 0 otherwise; and $z_{jkm} = 1$ if level j of attribute k is assigned to process m, 0 otherwise.

Expression (1) states the objective of maximizing the difference between net revenue – total revenue less loss due to cannibalization, and incremental fixed and variable costs. Disjunctive constraints (2) and (3) enforce the product selection criterion for each customer; the first term in (2) is the total utility U_i provided by the product to customer i. Constraint (4) requires each attribute to be specified at exactly one level. Constraints (5) and (6) guarantee that the selected level of each attribute is assigned to at least one process and the appropriate variable and fixed costs are taken into account. Finally, constraints (7) specify the nature of the variables. The following result indicates that it is unlikely that an efficient procedure for solving this problem exactly can be constructed.

Remark 1. **PDPS** *is NP-hard in the strong sense.*

In view of the computational complexity of **PDPS**, we construct a heuristic solution approach for solving it.

3 Heuristic Solution Procedure

The formulation of **PDPS** given by (1)–(7) indicates that two subproblems can be defined within it. The *Attribute- Level Selection Problem* is given by (1)–(4) and (7), and it requires specifying exactly one level for each attribute; the *Process Selection Problem* is defined by (1), (5)–(7), and it requires determining the manufacturing processes. x_{jk} and s_i are the coupling variables between these two subproblems. The proposed heuristic solution procedure exploits this structure by decomposing **PDPS** into the subproblems which are solved iteratively.

3.1 The Basic Algorithm

A brief statement of the algorithm is given below.

Algorithm **BasicDesign**

Step 1: Initialization: Determine initial product profile $\mathbf{X}^1$ by selecting appropriate values of $x_{jk}, k = 1, \ldots, K, \ j = 1, \ldots, J_k$. Set $n = 1, Z^0 = 0$.

Step 2: Process Selection Problem: For the given product $\mathbf{X}^n$ at iteration n, determine the optimal set of processes, the optimal set of switching customers and the optimal price. Let the set of processes selected be $\mathcal{M}^n$. Determine the profit Z^n corresponding to $\mathbf{X}^n$ and $\mathcal{M}^n$. Stop if $(Z^n - Z^{n-1})/Z^n \leq \epsilon$. Else, set $n \leftarrow n + 1$, and go to step 3.

Step 3: Attribute-Level Selection Problem: For the given set of processes $\mathcal{M}^{n-1}$, find the best product profile $\mathbf{X}^n$. Stop if $\mathbf{X}^n = \mathbf{X}^{n-1}$. Else, go to step 2.

The initial product profile can be determined in one of several ways. For example, any one of the existing products can serve as the starting profile. An approach to obtaining the initial solution that we found to be particularly effective in our computational studies was to solve the product design problem given in step 3 with all processes being available, i. e., $\mathcal{M}^0 = \mathcal{M}$. However, the performance of the algorithm depends more critically on solving Steps 2 and 3 efficiently. The solution approaches to these two problems are now discussed.

3.2 The Process Selection Problem

Consider the problem of determining the optimal set of processes for a given product **X**. Let j_k denote the level at which attribute $k, k = 1, 2, \ldots, K$ is active in this product. Its utility for customer i is $U_i = \sum_{k \in \mathcal{K}} w_{ij_k k}$. The resulting problem is

PSP

$$Z(\mathbf{X}) = \max \sum_{i \in \mathcal{I}} (\pi p_i - l_i) s_i - \sum_{m \in \mathcal{M}} f_m q_m - \sum_{i \in \mathcal{I}} p_i s_i \left(\sum_{m \in \mathcal{M}} \sum_{k \in \mathcal{K}} v_{j_k k m} z_{j_k k m} \right)$$

subject to

$$U_i - \pi - u_i^0 \leq B s_i \quad \forall i \tag{8}$$

$$\pi - U_i + u_i^0 \leq B(1 - s_i) \quad \forall i \tag{9}$$

$$\sum_{m \in \mathcal{M}} z_{j_k k m} \geq 1 \quad \forall k \tag{10}$$

$$z_{j_k k m} \leq q_m \quad \forall k, m \tag{11}$$

$$\pi \geq 0; s_i, z_{j_k k m}, q_m \in (0, 1) \quad \forall i, k, m \tag{12}$$

Now consider a restricted version of (**PSP**) with $\pi = \pi'$. The set of customers $\mathcal{I}_{\pi'}$ who switch to the new product is

$$\mathcal{I}_{\pi'} = \{ i | U_i - u_i \geq \pi' \}. \tag{13}$$

Let $P_{\pi'} = \sum_{i \in \mathcal{I}_{\pi'}} p_i$ represent the total population of the switching customers. The restricted problem can be written as

RPSP

$$Z(\mathbf{X}, \pi') = \max \left[\pi' P_{\pi'} - \sum_{i \in \mathcal{I}_{\pi'}} l_i \right] - \sum_{m \in \mathcal{M}} f_m q_m - P_{\pi'} \left(\sum_{m \in \mathcal{M}} \sum_{k \in \mathcal{K}} v_{j_k k m} z_{j_k k m} \right)$$

subject to

$$(10), (11), \text{ and } z_{j_k k m}, q_m \in \{0, 1\} \quad \forall k, m.$$

The first term in the objective is a constant with respect to the problem variables; it can, therefore, be ignored for the purpose of finding the optimal solution. The remaining problem has the structure of the uncapacitated facility location problem (UFL) if we treat attribute levels j_k as the unit demand points and processes m as candidate location sites. f_m is the fixed cost of opening a facility at location m, and $P_{\pi'} v_{j_k k m}$ is the cost of satisfying the demand at j_k from location m. Although UFL is NP-hard, efficient solution procedures are available for obtaining strong bounds and for solving reasonably large problems within acceptable computation time. In our computational study, we use the dual ascent procedure DUALOC developed by Erlenkotter (1978).

Let $Z'(\mathbf{X},\pi')$ denote the optimal solution to UFL, then

$$Z(\mathbf{X},\pi') = \pi' P_{\pi'} - \sum_{i \in \mathcal{I}_{\pi'}} l_i - Z'(\mathbf{X},\pi').$$

The solution to **PSP** is given by

$$Z(\mathbf{X}) = max_{\pi} Z(\mathbf{X},\pi) = Z(\mathbf{X},\pi^*).$$

Let $\delta_i = U_i - u_i^0$. The following result indicates that the search for π^* involves considering at most I values of π.

Proposition 1. $\pi^* \in \{\delta_1, \delta_2, \ldots, \delta_I\}$.

Consequently, at each iteration, we need to solve at most I uncapacitated facility location problems in order to determine the optimal set of processes. As shown in the proof of Proposition 1, this step generates the best price and the set of switching customers as well.

If we impose capacity limitations on individual processes then (11) is replaced by constraints of the form

$$P_{\pi'} z_{j_k km} \leq CAP_m q_m \quad \forall k, m$$

where CAP_m is the capacity of process m. In this case, the solution approach to **PSP** essentially requires solving a number of capacitated facility location (CFL) problems (Guignard and Opaswongkarn 1990), instead of the UFL problems considered here. While CFL is harder to solve optimally, several heuristic solution methods exist for it so that from an algorithmic standpoint, capacity constraints can be included within the solution approach.

3.3 The Attribute-Level Selection Problem

Now consider the problem for determining the optimal set of product attributes given a set of processes $\mathcal{M}^0 = \{m | q_m = 1\}$. Let

$$f(\mathcal{M}^0) = \sum_{m \in \mathcal{M}^0} f_m, \text{ and}$$

$$v'_{jk} = min_{m \in \mathcal{M}^0}\{v_{jkm}\} = v_{jkm'}.$$

Clearly, for a given (j,k), z_{jkm} equals 1 for $m = m'$, and equals 0 for all other m.

The attribute-level selection problem can now be stated as

ASP

$$Z(\mathcal{M}^0) = \max \sum_{i \in \mathcal{I}} \left[\left(\pi - \sum_{k \in \mathcal{K}} \sum_{j \in \mathcal{J}_k} v'_{jk} x_{jk} \right) p_i s_i - l_i s_i \right] - f(\mathcal{M}^0)$$

subject to

$$(2), (3)\ (4), \text{ and } s_i, x_{jk} \in \{0,1\} \quad \forall i,j,k.$$

ASP generalizes the model proposed by Kohli and Sukumar (1990) for the seller's return problem to include product price as an extrinsic variable. We construct a heuristic solution method (algorithm **BuildProduct**) for this problem. Similar to the notion of variable elimination used in non-serial dynamic programming (Chhajed and Lowe 1993), this procedure reduces one attribute at a time. Starting with attribute 1, this method successively performs a local optimization with respect to τ adjacent attributes that is followed by attribute augmentation. Here, τ is a parameter of the method; larger values of τ enable the procedure to consider more attributes simultaneously at the expense of increased computational time. If $\tau = 1$, this procedure reduces to an extension of the Kohli-Sukumar algorithm that incorporates product price. This procedure in now described; for simpler exposition, we assume that $\tau = 2$. The general case is discussed later in the section.

The heuristic method forms partial product profiles of increasing cardinality in the attribute space; the profile at the end of stage k comprises the first $k+2$ attributes. At the kth iteration in the procedure, only attributes k, $k+1$, and $k+2$ are considered. Let $J_t = |\mathcal{J}_t|$, $t = 1, \ldots, K$. For a pair of attribute levels $(j,q) \in \mathcal{J}_{k+1} \times \mathcal{J}_{k+2}$, J_k partial product profiles are constructed by associating each $r \in \mathcal{J}_k$ with (j,q). Of these J_k profiles, the profile (r_{jq}, j, q) that maximizes profit is selected. During attribute augmentation, partial product profile $\mathcal{L}(k+1,j;k+2,q) = \{(1,r_1^*),(2,r_2^*),\ldots,(k+1,j),(k+2,q)\}$ is generated which corresponds to attributes $1,\ldots,k+2$ where r_l^* is the "optimal" level for attribute l, $l = 1,\ldots,k$, and levels j and q are selected for attributes $k+1$ and $k+2$, respectively. Other quantities generated during attribute augmentation at the end of kth iteration are $w_i(k+1,j;k+2,q)$, $v(k+1,j;k+2,q)$, and $\Theta_{i,k+2}$. $w_i(k+1,j;k+2,q)$ can be interpreted as the total part worth of partial product $\mathcal{L}(k+1,j;k+2,q)$ for customer i, $v(k+1,j;k+2,q)$ is the variable cost of producing $\mathcal{L}(k+1,j;k+2,q)$, and $\Theta_{i,k+2}$ is the partial price of attributes $1,\ldots,k+2$ for i. The next iterative step involving attributes $k+1$, $k+2$, and $k+3$ is then solved with respect to these augmented attribute levels. A formal description of the procedure now follows.

Algorithm **BuildProduct**

Step 1: Initialization: For each customer i, determine the "partial price" of attribute k as

$$\theta_{ik} = \pi_i^0 U_{ik}^0 / U_i^0.$$

where U_{ik}^0 is the part worth of attribute k to customer i in the product that i buys currently. Set $k = 1$, and

$$\mathcal{L}(1,j;2,q) \quad = \quad \{(1,j),(2,q)\} \quad \forall j \in \mathcal{J}_1, \quad q \in \mathcal{J}_2$$

$$
\begin{aligned}
w_i(1,j;2,q) &= w_{ij1} + w_{iq2} \quad \forall i \in \mathcal{I},\ j \in \mathcal{J}_1,\ q \in \mathcal{J}_2 \\
v(1,j;2,q) &= v'_{j1} + v'_{q2} \quad \forall\ j \in \mathcal{J}_1,\ q \in \mathcal{J}_2 \\
\Theta_{i2} &= \theta_{i1} + \theta_{i2}, \quad \forall i \in \mathcal{I}.
\end{aligned}
$$

Step 2: Recursion: At the kth iteration, for each given level j of attribute $k+1$, and level q of attribute $k+2$

a. Local Optimization: determine the level r_{jq} of attribute k that maximizes the overall marginal contribution.

b. Attribute Augmentation: set

$$
\begin{aligned}
\mathcal{L}(k+1,j;k+2,q) &= \mathcal{L}(k,r_{jq};k+1,j) \bigcup \{(k+2,q)\} \\
w_i(k+1,j;k+2,q) &= w_{iq,k+2} + w_i(k,r_{jq};k+1,j), \quad \forall i \\
v(k+1,j;k+2,q) &= v'_{q,k+2} + v(k,r_{jq};k+1,j) \\
\Theta_{i,k+2} &= \theta_{i,k+2} + \Theta_{i,k+1}, \quad \forall i.
\end{aligned}
$$

Step 3: Termination: If $k \le K-2$, set $k = k+1$, and go to step 2. Else, select a product $\mathcal{L}(K-1,j^*;K,q^*)$ that gives the maximum profit. Set $x_{r_k^* k} = 1$ for $k = 1,\ldots,K$.

The major step in this algorithm involves local optimization at each stage. For a given level j of attribute $k+1$, and level q of attribute $k+2$ this problem can be formulated as:

$\mathbf{P1}(j,k+1;q,k+2)$

$$
r_{jq} = arg\ max_{r \in \mathcal{J}_k} \sum_{i \in \mathcal{I}} \left(p_i \left(\pi - v'_{q,k+2} - v(k,r;k+1,j)\right) - l_i\right) s_i \tag{14}
$$

subject to

$$
w_{iq,k+2} + w_i(k,r;k+1,j) - \pi - \left(U^0_{i,k+1} + U^0_{ik} - \Theta_{i,k+1} - \theta_{i,k+2}\right) \le Bs_i \tag{15}
$$

$$
\pi - \left(w_{iq,k+2} + w_i(k,r;k+1,j)\right) + \left(U^0_{i,k+1} + U^0_{ik} - \Theta_{i,k+1} - \theta_{i,k+2}\right) \le B(1-s_i) \tag{16}
$$

$$
s_i \in \{0,1\}\ \forall i; \quad \pi \ge 0.
$$

Equation (14) enforces local optimization with respect to attributes k, $k+1$, and $k+2$ while equations (15) and (16) insure that customer i will switch to the partial product if and only if it provides a higher surplus than the surplus offered by the current partial product.

To solve $\mathbf{P1}(j,k+1)$, we first find the optimal price π^*_{rjq} corresponding to each $r \in \mathcal{J}_k$. The maximum price that customer i is willing to pay to switch to the partial product defined by levels r, j and q for attributes k, $k+1$ and $k+2$, respectively, is given by

$$
e^i_{rjq} = w_{iq,k+2} + w_i(k,r;k+1,j) - \left(U^0_{i,k+1} + U^0_{ik} - \Theta_{i,k+1} - \theta_{i,k+2}\right).
$$

The profit corresponding to a price of e^i_{rjq} is given by

$$E_{irjq} = \sum_{g \in \mathcal{G}} \left[p_g \left(e^i_{rjq} - v'_{q,k+2} - v(k, r; k+1, j) \right) - l_g \right]$$

where $\mathcal{G} = \{g | g \in \mathcal{I}, e^g_{rjq} > e^i_{rjq}\}$. Arguments similar to those used in the proof of Proposition 1 yield

$$\pi^*_{rjq} \in \{e^1_{rjq}, \cdots, e^I_{rjq}\}.$$

Consequently, the maximal profit corresponding to level r of attribute k is given by

$$E^*_{rjq} = max_{i \in \mathcal{I}} \{E_{irjq}\}.$$

P1$(j, k+1; q, k+2)$ is then solved by

$$r_{jq} = arg \ max_{r \in \mathcal{J}_k} \{E^*_{rjq}\}.$$

For an arbitrary value of τ, $(1 \leq \tau \leq K)$, $w_i(\cdot)$, and $v(\cdot)$ are defined for attribute level combinations involving τ consecutive attributes. Note that when $\tau = K$, the algorithm finds the optimal solution by complete enumeration.

In **BuildProduct**, attribute k is eliminated at the kth iteration, and in order to do so, τ consecutive attributes $k+1, \ldots, k+\tau$ are considered. While the objective function (1), and constraints (2) and (3) enforce interdependence among the attributes, **BuildProduct** decomposes the problem and considers only a subset of these interdependencies. The graph shown in Figure 1 depicts the interdependencies considered by the algorithm when $\tau = 2$. Note that in order to eliminate attribute k, only *consecutive* attributes $k+1$ and $k+2$ are considered. This *dependency graph* (Chhajed and Lowe 1993) is a special case of a class of graphs called 2-trees[2]. It is easy to modify the above algorithm, following the approach in Chhajed and Lowe, for an arbitrary τ-tree dependency graph, in which the τ attributes considered are not consecutive. [Examples of 2-trees are given in Figure 2.] This methodology is useful when some *a priori* information about the nature of "correlation" between attributes is available. In addition, this approach can incorporate the case in which part worths reflect pairwise interaction among attribute levels.

INSERT FIGURES 1 AND 2 HERE

[2] A τ-*clique* is a complete graph on τ vertices. A τ-*Tree* is recursively defined as follows: A τ-clique is a τ-tree. Given a τ-tree, and a subgraph of the τ-tree that is a τ-clique, the graph obtained by introducing a new vertex, and connecting this vertex to each vertex of the τ-clique is again a τ-tree. The graph shown in Figure 1 can be constructed by first connecting nodes K and $K-1$. Next, node $K-2$ is connected to the end points of arc $(K-1, K)$. In the general case, when node k is introduced, it is connected to nodes $k+1$ and $k+2$. Within this framework, the Kohli-Sukumar algorithm uses a dependency graph that is a simple path connecting attribute k to $k+1$, $k = 1, \ldots, K-1$.

Thus algorithm **BasicDesign** essentially iterates between the product and the process spaces, thereby dealing separately with problems of reduced dimensionality. While it is difficult to analyze its convergence properties theoretically, in all our computational experiments, it was found to converge quite rapidly. However, there are instances in which it may converge to a solution that is only locally optimal. In order to overcome this drawback, we imbed this algorithm within a simulated annealing based approach. This enhancement is now described.

3.4 Algorithm Refinement Based on Simulated Annealing

Simulated annealing has been successfully employed for solving some difficult problems (see, for example, Johnson et al. 1989, Geman and Geman 1984, Kirkpatrick, Gelatt and Vecchi 1983, Matsuo, Suh and Sullivan 1987a, Matsuo, Suh and Sullivan 1987, Ahmadi and Tang 1991). We give below a brief description of this approach. For details, the interested reader is referred to Kirkpatrick et al. (1983), and Johnson et al. (1989).

Consider the following problem: max $h(y)$, subject to $y \in \mathcal{Y}$, where $h(y)$ is a real-valued function on domain $\mathcal{Y}$. Each element $y \in \mathcal{Y}$ has an associated set of *neighbors* $\mathcal{N}(y)$ such that any element $y' \in \mathcal{N}(y)$ can be reached from y by a one-step perturbation. A typical ascent algorithm based on neighborhood search moves from a given y to its neighbor y' if $h(y') > h(y)$. This approach will frequently yield solutions that are only locally optimal.

In contrast, simulated annealing permits occasional *downhill* moves, and thereby, provides a mechanism to escape from a local optima. The probability that a move will be made from y to y' is given by

$$\alpha = exp[-\{h(y) - h(y')\}^+/\text{TEMP}(g)]$$

where $[t]^+$ denotes $max(0, t)$, and TEMP(g) is a positive number. The *temperature* TEMP(g) at a given *state* g typically follows a geometric series given by $\text{TEMP}(g) = r * \text{TEMP}(g-1)$ where r is the *cooling rate* such that $0 < r < 1$. Under certain assumptions, it is possible to show (Anily and Federgruen 1985; Geman and Geman 1984) the existence of certain values of r that guarantee that simulated annealing will obtain the optimal solution in the limiting case.

For **PDPS**, we define the neighborhood of a given solution individually for the process and the product spaces. In the process space, the neighborhood $\mathcal{N}_1(\mathbf{Q}^r)$ of a given point $\mathbf{Q}^r = (q_1^r, \ldots, q_M^r)$ consists of all points $\mathbf{Q}^s = (q_1^s, \ldots, q_M^s)$ such that

$$\begin{aligned} q_m^s &= q_m^r \quad \text{for } m = 1, \ldots, M; \;\; m \neq n, \;\; and, \\ q_n^s &= 1 - q_n^r. \end{aligned}$$

It can be seen that the neighborhood of $\mathbf{Q}^r$ consists of M points. Similarly, the neighborhood $\mathcal{N}_2(\mathbf{X}^r)$ of a point $\mathbf{X}^r = (X_1^r, \ldots, X_K^r)$ in the product space comprises all points $\mathbf{X}^s$ such that

$$\begin{aligned} X_k^s &= X_k^r \quad \text{for } k = 1, \ldots, K \;\; k \neq n, \\ x_{jn}^s &= x_{jn}^r \quad \text{for } j = 1, \ldots, J_k \;\; j \neq g, \;\; and, \\ x_{gn}^s &= 1 - x_{gn}^r. \end{aligned}$$

It can be seen that the neighborhood of $\mathbf{X}^r$ consists of $\sum_{k=1}^{K}(J_k - 1)$ points. For a given pair $(\mathbf{X}^r, \mathbf{Q}^s)$ of points from the product and the process spaces, the optimum objective function value $Z(\mathbf{Y}^{rs})$ can be obtained as follows. For each (j,k) such that $x_{jk}^r = 1$, set $z_{jkm'} = 1$ where $m' = arg\ min_m\{v_{jkm} | q_m^s = 1\}$ is the machine with the minimum variable cost for producing attribute k at level j among all the open machines, and set all other $z_{jkm} = 0$. Next use the approach given in §3.2 to determine the optimal values of s_i and π. It follows, therefore, that a solution $\mathbf{Y}$ to **PDPS** is completely determined by the selection of the product profile and the set of open machines.

The parameters required for implementing this algorithm are the starting temperature INITTEMP; the cooling rate r; the termination threshold ϵ; and the maximum values N_1 and N_2 that the counters which control the number of neighbors scanned can take. A formal statement of the simulated annealing algorithm is given below. The various steps are described in detail subsequently. In the following, $Z(\cdot)$ represents the optimal objective function corresponding to solution $(\cdot)$, $\mathbf{Y}^{inc}$ denotes the incumbent solution, and Z_g is the best solution value obtained until temperature g.

Algorithm **IntegratedDesign**

Step 1: Initialization

Set the initial solution $\mathbf{Y}^0 = (\mathbf{X}^0, \mathbf{Q}^0)$ to the solution obtained from algorithm **BasicDesign**. Set the incumbent solution $\mathbf{Y}^{inc} = \mathbf{Y}^0$. Also set TEMP(0)=INITTEMP; $n_1 = 0$; $g = 0$; and $Z_0 = Z(\mathbf{Y}^0)$. Go to Step 2.

Step 2: Annealing

If $n_1 \geq N_1$, go to Step 5. Else, set $g = g - 1$; TEMP$(g) = r*$TEMP$(g+1)$; and $n_2 = 0$. Go to Step 3.

Step 3: Neighborhood Scan

a) Set $n_2 \leftarrow n_2 + 1$. If $n_2 < N_2$, fix $m = 1$, and go to Step 3b). Otherwise, set $Z_g = Z(\mathbf{Y}^{inc})$. If $Z_g \leq (1+\epsilon)Z_{g-1}$, then set $n_1 \leftarrow n_1 + 1$. Else, set $n_1 = 0$. Go to Step 2.

b) If $m > M$, go to Step 3a). Else, determine the neighbor $\mathbf{Q}^1$ of $\mathbf{Q}^0$ by setting $q_m^1 = 1 - q_m^0$. Determine $\mathbf{X}^1$ and $\mathbf{Y}^1$. Go to Step 4.

Step 4: Neighbor Selection

If $Z(\mathbf{Y}^1) > Z(\mathbf{Y}^0)$, set $\mathbf{Y}^0 = \mathbf{Y}^{inc} = \mathbf{Y}^1$, set $m \leftarrow m+1$, and go to step 3b). Otherwise, compute

$$\alpha = exp[-\{Z(\mathbf{Y}^1) - Z(\mathbf{Y}^0)\}/\text{TEMP}(g)].$$

Generate a random number a from the uniform distribution [0,1]. Set $\mathbf{Y}^0 = \mathbf{Y}^1$ if $\alpha \geq a$. Set $m \leftarrow m+1$. Go to step 3b).

Step 5: Reoptimization

Determine the optimal solution $\mathbf{Y}^*$ corresponding to the incumbent product profile $\mathbf{X}^{inc}$.

The algorithm is initialized with the solution obtained from **BasicDesign** in Step 1. Computational experience (see, for example, Johnson et al. 1989, and Ahmadi and Tang 1981) indicates that good starting solutions usually enhance the performance of simulated annealing. Step 2 implements the specified cooling schedule; it also determines whether the frozen state is reached. This algorithm maintains a counter n_1 that is incremented by one each time the incumbent solution value obtained at the end of the neighborhood search at any temperature does not exceed the incumbent solution value at the previous temperature by the threshold ϵ. The system is deemed to be frozen if the counter value equals N_1.

At each temperature, Step 3 controls the generation of neighbors. This is done by considering the neighborhood of the partial solution $\mathbf{Q}^0$ through N_2 cycles. In any cycle, the algorithm generates each of the M neighbors of $\mathbf{Q}^0$ in turn. For any neighbor $\mathbf{Q}^1$, the approach given in §3.3 is used to determine the corresponding product profile $\mathbf{X}^1$. Subsequently, $\mathbf{Y}^1$ is determined from $\mathbf{Q}^1$ and $\mathbf{X}^1$. Step 4 in the algorithm executes the logic pertaining to the acceptance of a neighboring solution, and if necessary, updates the current and the incumbent solutions. The solution obtained at the end of simulated annealing is reoptimized at Step 6. This is done by retaining the incumbent product profile $\mathbf{X}^{inc}$, and solving the process selection problem following the procedure given in §3.2 to determine the optimal processes, the optimal price and the optimal set of switching customers.

Several parameter values were tested for implementing the simulated annealing phase of algorithm **IntegratedDesign**. The values actually used in our computational study were $r = 0.90$; $N_1 = 5$; $N_2 = 10$; and INITTEMP$= 0.01 * Z_0$ where Z_0 is the solution value obtained from **BasicDesign**.

4 An Example Problem

We construct below an example problem that illustrates the sequential and integrated approaches. This problem is an adaptation of the instance described in Cook and Gill (1993) that deals with the redesign of an automobile that has not gained market acceptance. The market comprise three customer segments,

none of which buys the product currently. The problems identified with the current product are i) a shorter warranty compared to the competition, ii) poor road handling due to a soft front suspension, and iii) bumpy ride. In order to overcome the deficiencies, marketing and engineering personnel identify 3 attributes to be added to the current product - one for each problem identified above. These are

1. *Increased warranty* – this can be offered in 2 packages, covering 4 years and 6 years, respectively. This will require negotiating a contract with the dealers and providing additional training. The estimated fixed cost is $10,000, and the expected variable costs of the two packages are $60 and $100, respectively.

2. *Improved front suspension* – this can be achieved by providing either a stiffer spring or a modified strut. These alternatives will result in different road handling capabilities. An existing supplier A can supply the spring for $300 per unit, and the strut for $200 per unit. A new company B can be developed as a supplier for the strut for a development cost of $25,000, and the variable cost is expected to be $100 per unit. However, B does not have the capability of producing the spring.

3. *Ride Comfort* – this can be achieved with stronger shock absorbers that can be provided in two designs, SA1 and SA2, with different levels of ride comfort. These shock absorbers can be supplied by an existing supplier C for the variable costs of $150 and $250 corresponding to level 1 and 2, respectively. Alternatively, if developed, B can supply them for the variable costs of $100 and $250, respectively.

The variable cost of manufacture given the current design with the existing facilities is $19,950, and the total utility associated with this design is $20,000 for each segment. Table 1 summarizes the cost data and gives other customer-related details. In this table, D refers to the dealers, and all numbers other than the population values are in dollars.

The sequential solution in this instance is $\pi = 20,700$; $s_1 = s_3 = 1$; $q_A = q_C = q_D = 1$; $z_{21D} = z_{12A} = z_{23C} = 1$, and all other variables are zero. The solution value $Z = \$44,000$. Note that this solution does not call for developing supplier B. The integrated, and optimal, solution is $\pi = 20,550$; $s_1 = s_2 = s_3 = 1$; $q_A = q_B = q_D = 1$; $z_{21D} = z_{22B} = z_{23B} = 1$, and all other variables are zero. In this case is $Z = \$46,750$. Note that in this case, the product price is lower to accommodate customer 2. Also, attribute 2 is now offered at level 2 (strut) which enables B to become a supplier who additionally supplies SA2 at a cheaper price. Although the modified product results in less revenue, it is more than offset by the resulting cost savings.

5 Computational Experience

Our computational study addresses the performance of algorithm **IntegratedDesign** with respect to the optimal solution – this is done by varying the various problem parameters. A version of **BuildProduct** with

$\tau = 1$ is implemented. We also determine the sequential solution in each instance. In any problem instance, the sequential solution is obtained by first determining the product profile using algorithm **BuildProduct** with $v'_{jk} = 0$ for all j, k. These costs are taken into account at the subsequent stage when we solve the process selection problem to yield the optimal set of processes as well as the optimal price.

It can be seen that, while the sequential approach performs limited optimization with respect to the processes and the product price for the selected product profile, it lacks the integrated approach's ability to revise this profile subsequently. Thus, while the integrated approach is likely to give better solutions, the purpose of these experiments is to identify the extent of improvement possible. We use an experimental investigation for this purpose as the complexity of the interaction between the product design and the process selection decisions precludes a closed form solution of the model.

5.1 Experimental Details

We evaluate the impact of changes in eight parameters on the performance of **IntegratedDesign**. These parameters are i) the ratio $\rho = \mu_v / w_{ave}$ of the mean variable cost to the mean part worth, ii) the mean process fixed cost μ_f, iii) the coefficient of variation of process variable cost ζ_v, iv) the coefficient of variation of fixed cost ζ_f, v) the number of processes M, vi) the ratio π_{ave}/U_{ave} which is a measure of the surplus enjoyed by the customers currently, vii) the mean attribute-level part worth μ_w, and viii) the coefficient of variation of part worth ζ_w. These parameters are varied one at a time while the others are retained at their base values given in Table 2. In all experiments, the total number of customers I is fixed at 20; customer populations p_i are sampled from the uniform distribution [200, 600] and the resulting values are retained across all experiments. [Preliminary experiments revealed that varying the values of I and p_i resulted in little change in the overall results.] The number of attributes K is fixed at 5, and the number of levels is retained at 3 for each attribute for the reasons discussed later in this section. Also, the marginal contributions $l_i = 0, \ \forall i$.

In each problem instance, we generate $I/5$ competing products; the active attribute levels for these products are assigned randomly. These products are assigned prices sampled from the uniform distribution $[aU_{ave}, bU_{ave}]$, where U_{ave} is a measure of the average utility across all customers and across all products currently available. $U_{ave} = K w_{ave}$, where w_{ave} is the mean realized part worth given by

$$w_{ave} = \frac{\sum_i \sum_k \sum_j w_{ijk}}{IK\eta}$$

and $\eta(= 3)$ is the number of attribute levels considered in the problem instance. The best product and the resulting surplus u_i^0 for any customer i are determined from the profiles of the competing products and their prices, as well as the part worths w_{ijk}. We vary a and b in our experiments to yield various values of the mean price of competing products $\pi_{ave} = U_{ave}(a + b)/2$.

INSERT TABLES 3 THROUGH 10 HERE

The actual fixed costs f_m used for the various processes are obtained by sampling from a uniform distribution which has mean μ_f, and for which the upper and lower limits are derived from μ_f and ζ_f. Similarly, the actual variable costs v_{jkm} are obtained from a uniform distribution determined by μ_v and ζ_v, and the part worths are derived from μ_w and ζ_w.

M is considered at the seven levels of 3, 5, 7, 9, 11, 13, 15. ζ_f, ζ_v and ζ_w are tested at levels of 0.08, 0.16, 0.24, 0.32, 0.40, 0.48 and 0.56. [Note that the largest value that the coefficient of variation of a uniformly distributed nonnegative random variable can take is 0.58.] μ_f is considered at levels \$60000, \$120000, \$180000, \$240000, \$300000, \$360000, and \$420000. Mean fixed cost values higher than \$420000 result in many solutions with zero objective function values; consequently, they are not considered. We test ρ at the seven levels of 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, and 0.7. Finally, we generate problems for seven values of the ratio $\pi_{ave}/(U_{ave})$ — 0.60, 0.65, 0.70, 0.75, 0.80, 0.85 and 0.90.

10 instances of each scenario are generated randomly. For a given instance, the optimal solution value Z_{opt} is obtained by first explicitly enumerating all η^K profiles. For each of these profiles, the process selection problem is solved to determine the optimal set of processes and the optimal price corresponding to that profile, and consequently the profit value as well. The optimal solution for the overall problem is the best profit value across all these product profiles. For each problem instance, we compute the performance ratio $PR_{int} = (Z_{opt} - Z_{int})/(Z_{opt})$, where Z_{int} is the solution value obtained from **IntegratedDesign**. Table 1 through 8 report the average values of Z_{int}, Z_{opt}, and PR_{int} across all 10 instances; also reported in parentheses is the number of times the optimal solution was obtained in these 10 instances. For comparison purposes, we also report the solution value Z_{seq} and the corresponding performance ratio $PR_{seq} = (Z_{opt} - Z_{seq})/(Z_{opt})$ for the sequential approach.

The exponential growth in the number of product profiles η^K to be generated explicitly limits the size of problems that can be considered in the experiments. Hence, in order to keep computational effort within reasonable limits, K and η are fixed at 5 and 3, respectively. An alternative approach of evaluating the performance of **IntegratedDesign** through the use of upper bounds, based on Lagrangean relaxation, did not yield satisfactory results. We found that these bounds were quite weak even for small problems. The major reason for the weakness of these bounds appears to be the fact that the attribute-level selection problem has very little structure; it is essentially a general integer program (also see Dobson and Kalish 1993).

Tables 3 through 10 indicate that algorithm **IntegratedDesign** frequently finds the optimal solution. Across the 56 scenarios and the 560 problem instances, it is 1.3% less than the optimal solution on average; on the

other hand, the sequential approach is 30.1% less than the optimal solution on average. The worst performance ratio for the integrated solution is 7.7% (averaged over 10 problems). In general, **IntegratedDesign** is quite robust; with the possible exception of the impact of μ_f shown in Table 4, PR_{int} appears to be insensitive to variations in problem parameters. On the other hand, the performance of the sequential approach is not only inferior, but quite variable as well.

INSERT TABLES 3 THROUGH 10 HERE

6 Model Use for Decision Support

New product development requires coordinating various subsystems within an organization. Urban, Hauser and Dholakia (1987) propose an iterative procedure that goes through the stages of product specification, market evaluation and design refinement based on marketing, engineering and manufacturing inputs. Another product development framework utilizes a hierarchical bilevel model in which first preliminary product and process designs are generated based on aggregate, approximate values of various demand, cost and market-related parameters; these are subsequently refined at the next level with more accurate and detailed estimates of these parameters (see, for example, Vonderembse and White 1991). Final product and process designs are determined after a number of iterations through these two stages. In both paradigms, the proposed approach provides a tool that has rapid modeling capability, and can generate good first-cut solutions together with the ability to do "what if" analysis.

SENSITIVITY ANALYSIS

We note that while the integrated approach is conceptually superior to the sequential approach for the reasons discussed earlier, its implementation is not straightforward. The coordination of personnel from different functional areas in order to obtain a consensus on product and process specifications is difficult to achieve in firms that are accustomed to the traditional sequential approach, and is quite likely to entail additional cost. In such cases, it is important for the firm to be able, at early stages of the iterative product development cycle such as one proposed by Urban et al., to estimate the marginal benefit obtained by adopting the integrated approach. Clearly, if this benefit is outweighed by the additional cost likely to be incurred, then from that stage of iteration onwards the firm should switch to the sequential approach.

The computational study described in §4 illustrates how algorithm **IntegratedDesign** provides this capability of doing sensitivity analysis. For the specific example considered, Table 7 indicates that if the number of alternative processes $M = 15$, then the marginal benefit of adopting the integrated approach (using **IntegratedDesign**) is $47,000 which may not be large enough to cover the additional cost. On the other hand, if $M = 3$, then the marginal benefit of $238,000 may well be worth it.

Fixed Product Development Cost

Dobson and Kalish (1993) note the incidence of the fixed development cost, say F_D, in new product design. Let $F_D = F_{D0} + \sum_k F_{Dk}$, where F_{Dk} denotes the fixed development cost resulting from attribute k, and F_{D0} is the residual. Furthermore, let $F_{Dk} = F_{Dk0} + \sum_j F_{Dkj}$, where F_{Dkj} is the element of attribute fixed cost that can be ascribed to level j, and F_{Dk0} is the residual within attribute k. Note that $F_{D0} + \sum_k F_{Dk0}$ is a sunk cost once the product development is undertaken, and it can be ignored for both product design and process selection decisions. F_{Dkj} can be accounted for in the model by modifying the last term in (1) to read

$$\sum_k \sum_j \sum_m \left(\sum_i p_i s_i v_{jkm} + F_{Dkj} \right) z_{jkm}$$

This leaves the solution procedure essentially unchanged. Note that it is possible to incorporate this modification because solutions in which $\sum_m z_{jkm} \leq 1$ are dominant.

Accounting for Existing Products and Processes

Consider the case in which the firm manufactures other products that the proposed new product will compete with. At the end of the algorithm **IntegratedDesign**, we obtain the set $\mathcal{I}_s = \{i | s_i = 1, l_i > 0\}$ of current customers who will switch to the new product. The firm then needs to re-evaluate the profitability of the existing products in view of the possible loss of demand due to the switching customers.

It is possible that some of the processes currently existing within the firm can be used for producing one or more attributes at various levels. An existing process m is accounted for by putting $f_m = 0$, and using the relevant v_{jkm}. This enables the firm to decide, say, between using an existing equipment that is relatively inefficient (resulting in higher variable costs) and buying a new one.

Cost Estimation

In this model, we need to specify the fixed costs for each process, and the variable costs for each process and attribute level combination. While it is difficult to extract relevant cost data in most real cases, the fixed cost of a process is usually relatively easy to determine. This consists of the purchase or leasing cost, cost of maintenance contracts and periodic preventive maintenance, cost of accessories such as jigs, fixtures, dies, and other material handling equipment. For automated workcenters, the software costs need to be included as well. These costs are derived from vendor contracts and from the firm's records. Attribute level- and process-specific variable costs include direct material, direct labor, power, consumables, etc. that can be directly related to each unit manufactured. These can be estimated for known processes from existing cost records (Green and Kreiger 1992); for new processes, they can be estimated from the data provided by equipment manufacturers if the process represents a manufacturing facility, or from supplier quotations if the process denotes a supplier (also see Dobson and Kalish 1993 for a related discussion).

Development costs are more difficult to capture. These include an allocation of the salaries of the R&D personnel, cost of procuring samples, equipment and other special items related to the new product, laboratory and testing expenses, cost of CAD and other software, etc. As we noted earlier, once the product development decision is undertaken, we need consider only F_{Djk}, i.e., the element of fixed cost that varies across attribute levels. Many companies follow the practice of designing different subassemblies of the product in parallel with each subassembly handled by a different design team. Thus, for an automobile, engine design can occur in parallel with the design of the gear train, body styling, etc. This facilitates the breakdown of total development costs into attribute specific costs.

7 Summary and Enhancements

This paper models an integrated framework in which the decision on new product design is made jointly with the selection of optimal processes required to manufacture the product. This problem is formulated as nonlinear integer program. We construct a solution approach that has the conceptually appealing property of decomposing the problem into the product design and process selection subproblems that represent the marketing and the manufacturing aspects of this decision, while providing a mechanism to link these two subproblems together. Our emphasis in this paper is on constructing a basic model for the integrated approach. As discussed elsewhere in this paper, algorithmic extensions within such an approach are possible; it is also possible, at a cost, to relax some of its assumptions.

The significance of product-process interactions at the system design level has been emphasized in the literature on concurrent engineering, simultaneous engineering, and quality functional deployment. We believe that this study is one of the earliest attempts at modeling these interactions and proposing an integrated solution approach. This model and the solution approach can be used to answer questions such as:

- Should a new product be introduced at all?
- Is it better to buy a flexible machine that is capable of providing a number of attributes or should dedicated machines be purchased?
- Should a new option, favored by a large number of customers, be offered even if it requires significant fixed expenditure?
- If a company is thinking about expanding its current line of products, which of the attribute levels currently being offered should be duplicated in order to minimize incremental fixed costs?

Acknowledgements

The authors thank Don Erlenkotter for providing the DUALOC code, and Marilyn Lucas and Sanjeev Sinha for the extensive computational support. The paper has benefitted from the comments of Edward F. McDonough III and two anonymous referees on an earlier draft. This research was partly supported by a grant from the Research Board at the University of Illinois.

References

1. Ahmadi, R. H. and C. S. Tang (1991), "An Operation Partitioning Problem for Automated Assembly System Design," *Operations Research*, Vol. 39, 824–835.

2. Albers, S. (1976), "A Mixed Integer Nonlinear Programming Procedure for Simultaneously Locating Multiple Products in Attribute Space," in *Methods of Operations Research*, edited by R. Henn et al., Proceedings of the First Symposium on Operational Research, Univ. of Heidelberg, Germany, 899–909.

3. Anily, S. and A. Federgruen (1985), "Simulated Annealing Methods with General Acceptance Probabilities," Working Paper, Graduate School of Business, Columbia University, New York, NY.

4. Brazier, D. and M. Leonard (1990), "Concurrent Engineering: Participating in Better Designs," *Mechanical Engineering*, January, 52–53.

5. Boothroyd, G. (1988), "Design for Assembly," *Mechanical Engineering*, February, 28–31.

6. Boothroyd, G. and P. Dewhurst (1988), "Product Design for Manufacturing and Assembly," *Manufacturing Engineering*, April.

7. Chakravarty, A. K. and J. E. Baum (1992), "Co-ordinated Planning for Competitive Products and Their Manufacturing Operations," *International Journal of Production Research*, Vol. 30, 2293–2311.

8. Chhajed, D. and T. J. Lowe (1993), "Solving Structured Multifacility Location Problems Efficiently," *Transportation Science*, forthcoming.

9. Cook, H. E. (1993), *Personal Communication.*

10. Cook, H. E. and M. R. Gill (1993), "On System Design," *Research in Engineering Design: Theory, Applications and Concurrent Engineering*, Vol. 4, 215–226.

11. Corneujols, G., G. L. Nemhauser and L. A. Wolsey (1990), "The Uncapacitated Plant Location Problem," in *Discrete Location Theory*, edited by R. L. Francis and P. Mirchandani, John Wiley and Sons, New York, NY.

12. Dobson, G. and S. Kalish (1988), "Positioning and Pricing a Product Line," *Marketing Science*, Vol. 7, 107–125.

13. Dobson, G. and S. Kalish (1993), "Heuristics for Pricing and Positioning a Product-Line Using Conjoint and Cost Data," *Management Science*, Vol. 39, 160–175.

14. Eliashberg, J. and A. K. Manrai (1989), "Optimal Positioning of New Products: Some Analytical Implications and New Results," Working Paper No. 89-003, Wharton School, University of Pennsylvania, Philadelphia, PA.

15. Erlenkotter, D. (1978), "A Dual-Based Procedure for Uncapacitated Facility Location," *Operations Research*, Vol. 26, 992–1009.

16. Eriksen, S.E. and P. D. Berger (1987), "A Quadratic Programming Model for Product Configuration Optimization," *Zeitschrift für Operations Research*, Vol. 31, B143–B159.

17. Evans, B. (1988), "Simultaneous Engineering," *Mechanical Engineering*, February, 38–42.

18. Garey, M. R. and D. S. Johnson (1979), *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman and Company, New York, NY.

19. Gavish, B., D. Horsky and K. N. Srikanth (1983), "An Approach to Optimal Positioning of a New Product," *Management Science*, Vol. 29, 1277–1297.

20. Geman, S. and D. Geman (1984), "Stochastic Relaxation, Gibbs Distribution, and the Bayesian Restoration of Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 6, 721–741.

21. Green, P. E. and A. M. Kreiger (1985), "Models and Heuristics for Product Line Selection," *Marketing Science*, Vol. 4, 1–19.

22. Green, P. E. and A. M. Kreiger (1992), "An Application of Product Positioning Model to Pharmaceutical Products," *Marketing Science*, Vol. 11, 117–132.

23. Green, P. E., J. D. Carroll and S. M. Goldberg (1987), "A General Approach to Product Design Optimization via Conjoint Analysis," *Journal of Marketing*, Vol. 45, 17–37.

24. Guignard, M. and K. Opaskongworn (1990), "Lagrangian Dual Ascent Algorithms for Computing Bounds in Capacitated Plant Location Problems," *European Journal of Operational Research*, Vol. 46, 73–83.

25. Hauser, J. R. and P. Simmie (1981), "Profit Maximizing Perpetual Positions: An Integrated Theory for the Selection of Product Features and Price," *Management Science*, Vol. 27, 33–56.

26. Hauser, J. R. and D. Clausing (1988), "The House of Quality," *Harvard Business Review*, May–June, 63–73.

27. Johnson, R. M. (1974), "Tradeoff Analysis of Consumer Values," *Journal of Marketing Research*, Vol. 11, 121–127.

28. Johnson, D. S., C. R. Aragon, L. A. McGeoch and C. Schevon (1989), "Optimization by Simulated Annealing: An Experimental Evaluation, Part I, Graph partitioning," *Operations Research*, Vol. 37, 865–892.

29. Kirkpatrick, S., C. D. Gelatt, Jr. and M. P. Vecchi (1983), "Optimization by Simulated Annealing," *Science*, Vol. 220, 671–680.

30. Kohli, R. and R. Krishnamurti (1987), "A Heuristic Approach to Product Design," *Management Science*, Vol. 33, 1123–1133.

31. Kohli, R. and R. Sukumar (1990), "Heuristics for Product- Line Design," *Management Science*, Vol. 36, 1464–1478.

32. Matsuo H., C. J. Suh and R. S. Sullivan (1987a), "A Controlled Search Simulated Annealing Method for the Single Machine Weighted Tardiness Problem," Working Paper #87-12-2, Graduate School of Business, Univ. of Texas, Austin, TX.

33. Matsuo H., C. J. Suh and R. S. Sullivan (1987b), "A Controlled Search Simulated Annealing Method for General Job Shop Scheduling Problem," Working Paper, Graduate School of Business, Univ. of Texas, Austin, TX.

34. McBride, R. D. and F. S. Zufreyden (1988), "An Integer Programming Approach to the Optimal Product Line Selection Problem," *Marketing Science*, Vol. 7, 126–140.

35. Page, A. L. and H. F. Rosenbaum (1987), "Redesigning Product Lines with Conjoint Analysis: How Sunbeam Does it," *Journal of Product Innovation Management*, Vol. 4, 120–137.

36. Shocker, A. D. and V. Srinivasan (1974), "A Consumer-Based Methodology for the Identification of New Product Ideas," *Management Science*, Vol. 20, 921–937.

37. Sudharshan, D., J. H. May and A. D. Shocker (1987), "A Simulation Comparison of Methods for New Product Location," *Marketing Science*, Vol. 6, 182–201.

38. Stoll, H. W. (1988), "Design for Manufacture," *Manufacturing Engineering*, Vol. 100–1, 67–73.

39. Sudharshan, D., J. H. May and T. S. Gruca (1988), "DIFFSTRAT: An Analytical Procedure for Generating New Product Concepts for a Differentiated-Type Strategy," *European Journal of Operational Research*, Vol. 36, 50–65.

40. Suri, R. and M. Shimizu (1989), "Design for Analysis: A New Strategy to Improve Design Process," *Research in Engineering Design*, Vol. 1, 105–120.

41. Ulrich, K., D. Sartorius, S. Pearson and M. Jakiela (1993), "Including the Value of Time in Design-For-Manufacturing Decision making," *Management Science*, Vol. 39, 429–447.

42. Urban, G. L., J. R. Hauser and N. Dholakia (1987), *Essentials of New Product Management*, Prentice-Hall, Englewood Cliffs, NJ.

43. Vonderembse, M. A. and G. P. White (1991), *Operations Management: Concepts, Methods and Strategies*, West Publishing Company, St. Paul, MN.

44. Winter, F. W. (1993), *Personal Communication.*

45. Zufreyden, F. S. (1977), "A Conjoint-Measurement-Based Approach for Optimal New Product Design and Product Positioning," in *Analytical Approaches to Product and Market Planning*, edited by A. D. Shocker, Marketing Science Institute, Cambridge, MA.

46. Zufreyden, F. S. (1979), "ZIPMAP-A Zero-One Integer Programming Model for Market Segmentation and Product Positioning," *Journal of the Operational Research Society*, Vol. 30, 63–76.

Appendix 1

Proofs

Remark 1. **PDPS** *is NP-hard in the strong sense.*

PROOF: The result is proved by restriction: we show that a special case of **PDPS** is NP-hard in the strong sense. Consider the case in which $I = 1$, $l_1 = 0$, $p_i = 1$, $u_1^0 = 0$, $J_k = 1$, and $w_{11k} = 1$, $\forall k$. Clearly, in an optimal solution, $\pi = K$, and $s_1 = 1$. **PDPS** then reduces to

$$Z = \max \quad K - \sum_{m \in \mathcal{M}} f_m q_m - \sum_{m \in \mathcal{M}} v_{1km} z_{1km}$$

subject to

$$\sum_{m \in \mathcal{M}} z_{1km} \geq x_{1k} \quad \forall k$$

$$z_{1km} \leq q_m \quad \forall k, m$$

$$z_{1km}, q_m \in (0, 1) \quad \forall k, m$$

This problem has the structure of the uncapacitated facility location (UFL) problem [see, for example, Erlenkotter 1978]. The result follows from the observation that the vertex cover problem that is known to be NP-hard in the strong sense (see, for example, Garey and Johnson 1979) can be transformed to UFL (Cornuejols, Nemhauser and Wolsey 1990). □.

Proposition 1. $\pi^* \in \{\delta_1, \delta_2, \ldots, \delta_I\}$.

PROOF: Order δ_i such that $\delta_{[1]} \leq \delta_{[2]} \leq \cdots \leq \delta[I]$, and let $\delta_{[0]} = 0$. When $\pi = \delta_{[i]}$, it follows from (13) that $s_j = 1$ for $j \in \{[i], [i+1], \ldots, [I]\}$, and $s_j = 0$ for all other j. s_j remains unchanged for any value of π in the interval $(\delta_{[i-1]}, \delta_{[i]}]$. Consequently,

$$z(\mathbf{X}, \pi) \leq z(\mathbf{X}, \delta_i) \quad \text{for } \pi \in (\delta_{[i-1]}, \delta_{[i]}].$$

Repeating this argument for each of I such intervals yields the desired result. □

TABLE 1 – Example Problem Data

	$k=1$ (*Warranty Period*)		2 (*Front Suspension*)		3 (*Ride Comfort*)		
	$j=1$ (*4 years*)	2 (*6 years*)	1 (*Spring*)	2 (*Strut*)	1 (*SA1*)	2 (*SA 2*)	
Customers							
w_{1jk}	500	550	200	180	300	450	$p_1 = 300$
w_{2jk}	700	950	100	150	400	450	$p_2 = 5$
w_{3jk}	450	500	300	50	300	650	$p_3 = 240$
Processes							
v_{jkA}	∞	∞	300	200	∞	∞	$f_A = 0$
v_{jkB}	∞	∞	∞	100	100	250	$f_B = 25000$
v_{jkC}	∞	∞	∞	∞	150	300	$f_C = 0$
v_{jkD}	60	100	∞	∞	∞	∞	$f_D = 10000$

TABLE 2

Base Parameter Values

Parameter	*Base Value*
μ_w	200
ζ_w	0.4
ρ	0.4
μ_f (\$)	60K
ζ_v	0.4
ζ_f	0.4
K	5
η	5
M	5
π_{ave}/U_{ave}	0.8

TABLE 3 – Impact of the Ratio of Average Variable Cost to Average Parts Worth

ρ	*Solution Value*			*Relative Performance*	
	Z_{seq} (million \$)	Z_{int} (million \$)	Z_{opt} (million \$)	PR_{seq}	PR_{int}
0.1	1.687	1.704	1,732	0.974 (3)	0.983 (2)
0.2	1.356	1.445	1.471	0.920 (2)	0.983 (4)
0.3	1.028.	1.205	1.230	0.836 (0)	0.979 (6)
0.4	0.868	1.085	1.094	0.791 (1)	0.991 (8)
0.5	0.535	0.889	0.890	0.594 (1)	0.999 (9)
0.6	0.356	0.697	0.722	0.478 (1)	0.962 (5)
0.7	0.222	0.540	0.553	0.402 (0)	0.968 (6)

TABLE 4 – Impact of Average Fixed Cost

μ_f ('000 \$)	*Solution Value*			*Relative Performance*	
	Z_{seq} (million \$)	Z_{int} (million \$)	Z_{opt} (million \$)	PR_{seq}	PR_{int}
60	0.764	0.993	0.993	0.770 (0)	1.000 (10)
120	0.664	0.878	0.885	0.752 (0)	0.992 (7)
180	0.616	0.825	0.835	0.742 (0)	0.987 (7)
240	0.556	0.732	0.745	0.747 (0)	0.978 (9)
300	0.530	0.671	0.718	0.731 (0)	0.923 (7)
360	0.469	0.650	0.682	0.690 (0)	0.946 (7)
420	0.396	0.593	0.638	0.604 (0)	0.925 (7)

TABLE 5 – Impact of Coefficient of Variation of Variable Cost

ζ_v ('000 \$)	Solution Value			Relative Performance	
	Z_{seq} (million \$)	Z_{int} (million \$)	Z_{opt} (million \$)	PR_{seq}	PR_{int}
0.08	0.313	0.530	0.538	0.584 (0)	0.985 (6)
0.16	0.346	0.641	0.647	0.533 (0)	0.990 (7)
0.24	0.479	0.740	0.745	0.644 (0)	0.992 (8)
0.32	0.714	0.925	0.930	0.765 (0)	0.994 (9)
0.40	0.878	1.062	1.070	0.821 (0)	0.991 (7)
0.48	1.027	1.240	1.246	0.828 (0)	0.996 (9)
0.56	1.270	1.456	1.467	0.872 (2)	0.992 (7)

TABLE 6 – Impact of Coefficient of Variation of Fixed Cost

ζ_f ('000 \$)	Solution Value			Relative Performance	
	Z_{seq} (million \$)	Z_{int} (million \$)	Z_{opt} (million \$)	PR_{seq}	PR_{int}
0.08	0.767	0.993	0.993	0.771 (0)	1.000 (10)
0.16	0.760	0.982	0.982	0.774 (0)	1.000 (10)
0.24	0.774	1.001	1.001	0.773 (0)	1.000 (10)
0.32	0.773	1.001	1.001	0.773 (0)	1.000 (10)
0.40	0.763	0.984	0.984	0.775 (0)	1.000 (10)
0.48	0.761	1.002	1.002	0.761 (0)	1.000 (10)
0.56	0.754	0.964	0.964	0.783 (0)	1.000 (10)

TABLE 7 – Impact of the Number of Potential Processes Available

M ('000 \$)	*Solution Value* Z_{seq} (million \$)	Z_{int} (million \$)	Z_{opt} (million \$)	*Relative Performance* PR_{seq}	PR_{int}
3	0.708	0.941	0.952	0.744 (1)	0.990 (7)
5	0.769	0.962	0.987	0.776 (0)	0.975 (6)
7	0.768	1.025	1.082	0.709 (0)	0.945 (2)
9	1.132	1.274	1.274	0.883 (4)	1.000 (10)
11	1.079	1.251	1.256	0.856 (3)	0.995 (9)
13	1.185	1.240	1.246	0.950 (4)	0.995 (7)
15	1.176	1.223	1.230	0.954 (6)	0.997 (8)

TABLE 8 – Impact of the Ratio of Average Competitive Price to Average Parts Worth

π_{ave}/U_{ave} ('000 \$)	*Solution Value* Z_{seq} (million \$)	Z_{int} (million \$)	Z_{opt} (million \$)	*Relative Performance* PR_{seq}	PR_{int}
0.60	0.926	1.399	1.439	0.646 (1)	0.971 (7)
0.65	0.767	1.120	1.182	0.653 (1)	0.951 (4)
0.70	0.727	1.021	1.047	0.693 (0)	0.971 (6)
0.75	0.492	0.884	0.991	0.541 (0)	0.967 (4)
0.80	0.452	0.849	0.860	0.502 (0)	0.985 (7)
0.85	0.397	0.695	0.711	0.537 (0)	0.977 (8)
0.90	0.239	0.570	0.593	0.380 (0)	0.961 (4)

TABLE 9 – Impact of Mean Parts Worth

μ_w	Solution Value			Relative Performance	
	Z_{seq} (million \$)	Z_{int} (million \$)	Z_{opt} (million \$)	PR_{seq}	PR_{int}
50	0.090	0.163	0.168	0.534 (0)	0.971 (7)
100	0.205	0.390	0.402	0.489 (0)	0.969 (7)
150	0.335	0.607	0.619	0.533 (0)	0.981 (7)
200	0.616	0.798	0.805	0.752 (1)	0.991 (7)
250	0.824	1.088	1.106	0.731 (1)	0.984 (6)
300	1.074	1.421	1.443	0.744 (1)	0.985 (7)
350	1.184	1.540	1.570	0.754 (2)	0.980 (6)

TABLE 10– Impact of the Coefficient of Variation of Parts Worth

ζ_w	Solution Value			Relative Performance	
	Z_{seq} (million \$)	Z_{int} (million \$)	Z_{opt} (million \$)	PR_{seq}	PR_{int}
0.10	0.482	0.804	0.814	0.594 (0)	0.988 (5)
0.20	0.220	0.666	0.674	0.325 (0)	0.988 (7)
0.30	0.382	0.742	0.756	0.507 (0)	0.981 (5)
0.40	0.588	0.730	0.748	0.778 (1)	0.976 (4)
0.50	0.484	0.747	0.766	0.617 (0)	0.975 (4)
0.60	0.725	0.938	0.951	0.767 (1)	0.986 (7)
0.70	0.840	1.039	1.096	0.756 (1)	0.948 (3)

Number of Attributes = 6

Figure 1: Dependency Graph Considered by BuildProduct with τ=2

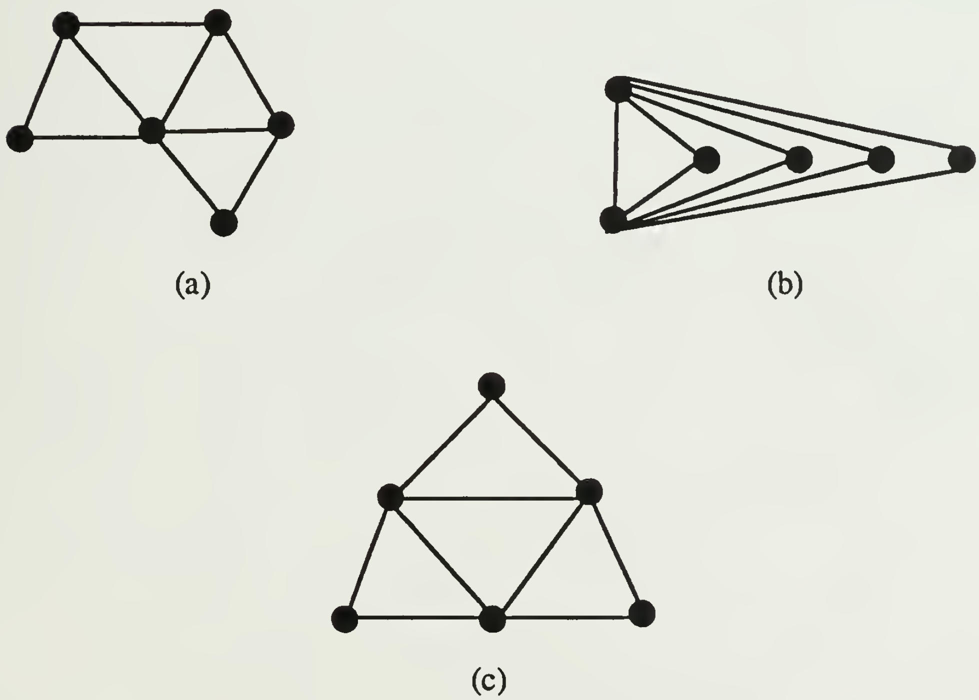

Figure 2. Examples of 2-Tree